Sultan Qaboos University
College of Economics and Political Science
Department of Information Systems
INFS6212 Data Analysis for Business

Project

Telecommunication Customer Churn Prediction

Names: Asma Alshahri - 143416
Shaima Alhsani – 143415

Course: INFS6212

Submitted to: Dr. Shaikh

Table of Contents

Introduction ***4***

Brief about the Data Set ***5***

Steps to apply Algorithms ***7***

1. Data Preprocessing **8**

1.1 Check Structure and Missing Values 8

1.2 Show Missing Values as a Data Frame 8

1.3 Visualize Missing Data 9

1.4 Data cleaning 9

2. Exploratory Data Analysis (EDA) **9**

2.1 Visualizing the Correlation 10

2.1.1 Visualizing categorical Features: 10

- **Churn by Contract Type** 10
- **Churn by Internet Services** 11

2.1.2 Visualizing Numerical Features 11

- **Tenure** 12
- **Monthly Charges** 12
- **Total Charges** 13

3. Applying Three Machine Learning Algorithms for Churn Prediction **13**

3.1 Data Preparation 14

3.2 Machine Learning Model Evaluations and Predictions: 15

1. Logistic Regression 15
2. XGBoost 16
3. Support Vector Machine (SVM) 17

Discussion and Results ***18***

Interpretation of Results 19

Business Implications 19

Limitations 21

Recommendations 21

Conclusion ***22***

References ***23***

Appendix A ***24***

Loading libraries 24

Explore Data Structure 24

Visualizing the Correlation 24

Visualize Missing Data 24

Data cleaning 25

Encoding Categorical Variables 25

Select Numeric Variables 25

Colleration matrix 25

Churn by Contract Type 26

Churn by Contract Type 26

Churn by Internet Services 26

Monthly charge by Churn 26

Total charge by Churn 26

Tenure by Churn 26

Applying Algorithm: 27

4.1.1 load the essential R packages: 27

4.1.2 Convert Churn to a factor for classification: 27

4.1.3 Train-Test Split 27

4.1.4 Define Heatmap Plot Function 27

Logistic Regression.....28
XGBoost.....29
SVM.....30

Lost of Figures

Figure 1: Distribution of the Target Variable (Churn).....6
Figure 2: Project workflow.....7
Figure 3: Checking Data Structure and Identifying Missing Values in the Dataset.....8
Figure 4: Missing Value in Dataset.....8
Figure 5: Visualization of missing data in the dataset.....9
Figure 6: Churn rate by type of contract.....10
Figure 7:Churn rate by Internet services.....11
Figure 8: Density Plot of Tenure by Churn.....11
Figure 9: Density Plot of Monthly Charges by Churn.....12
Figure 10: Density Plot of Total Charges by Churn.....13

List of Tables

Table 1: Description of variables in Telecom Churn Dataset.....5
Table 2: Logistic Regression: Confusion Matrix Output and Heatmap Visualization.....14
Table 3: XGBoost: Confusion Matrix Output and Heatmap Visualization.....15
Table 4: Support Vector Machine (SVM): Confusion Matrix Output and Heatmap Visualization.....16
Table 5: Performance Comparison of Machine Learning Models for Customer Churn Prediction.....17

Introduction

Customer churn is defined as the termination of subscription to service by existing customers, poses a persistent challenge for companies in subscription-based industries, particularly telecommunications. Previous Studies have shown that gaining new customers is considerably costlier than retaining existing ones (Hadden et al., 2007). Telecom companies typically face significant churn rates. For example, towards the end of 2021, the average churn rate was reported to be approximately 31%, compared to a lower average of 21% just a year earlier in 2020 (Tessitore, 2024). Consequently, predicting customer churn accurately is crucial for telecom operators aiming to implement proactive customer retention strategies.

This project utilizes the Telco Customer Churn dataset, a publicly available dataset hosted on Kaggle, which simulates a real-world customer base for a fictional telecom company. It contains detailed records for 7,043 customers, each described by 21 features covering demographics, service subscriptions, billing behavior, and a target variable indicating whether the customer has churned. The dataset includes both categorical and numerical variables, and like many business datasets, it presents challenges such as class imbalance and missing values.

To explore and evaluate predictive models for customer churn, this project applies three supervised machine learning algorithms:

- Logistic Regression is a widely used linear classifier that is both interpretable and efficient for binary classification tasks.
- Support Vector Machine algorithm is a classification method that does not rely on probabilities and identifies an optimal hyperplane to separate churners from non-churners, suitable for handling complex data patterns.

- Extreme Gradient Boosting algorithm (XGBoost) is a high-performance, decision-tree-based ensemble method that often achieves state-of-the-art accuracy on structured data tasks (Chen & Guestrin, 2016).

The performance of each model is evaluated using standard classification metrics, including accuracy analysis, confusion matrix-based, and recall. The goal is to determine which algorithm most effectively identifies customers at risk of churning, offering actionable insights to support retention strategies in a telecom context.

Brief about the Data Set

The dataset utilized in this project is the Telco Customer Churn dataset, originally developed by IBM and publicly accessible via Kaggle.com. It contains 7,043 records representing customers of a fictional telecommunications company. Each record is described by 21 variables, covering customer demographics, service subscriptions, billing information, and churn status.
Key categories of information included in the dataset are:

- **Churn status:** The "Churn" variable indicates whether a customer discontinued their service during the previous month.

- **Service Subscriptions**: outline the specific services to which each customer is subscribed, including phone service, type of internet, technical support service, online security service, backup services, device protection service, and streaming options.

- **Account details:** Contains customer-specific account data, including the duration of their subscription (tenure), type of contract, chosen payment method, use of paperless billing, monthly fees, and the total amount paid.

- **Customer Demographics:** Covers personal information such as gender, senior citizen status, and whether the individual has a partner or any dependents.

A complete summary of all variables is presented in the following table:

Table 1: Description of variables in Telecom Churn Dataset

	Variable	Description	Data Type
1	Customer_ID	A unique identifier is assigned to each customer	Character
2	Gender	This indicates the gender of the customer	Categorical
3	Senior_Citizen	Indicate whether the customer is a senior citizen or not (Yes, No)	Numeric (binary)
4	Partner	Specify whether the customer has a partner or not.	Categorical
5	Dependents	Specify whether the customer has any dependents.	Categorical
6	Tenure	In a month's duration of the customer's tenure with the company	Integer
7	Phone_Service	Specify whether the customer has a phone service or not	Categorical
8	Multiple_Lines	Indicates whether the customer has multiple lines or not	Categorical
9	Internet_Service	Type of internet connection the customer uses.	Categorical
10	Online_Security	Specify the customers' subscription to online security.	Categorical
11	Online_Backup	Indicates if online backup services are subscribed to.	Categorical
12	Device_Protection	Shows if device protection is part of the customer's plan.	Categorical
13	Tech_Support	Whether customer has technical support.	Categorical
14	Streaming_TV	Indicates access to streaming television services.	Categorical
15	Streaming_Movies	Whether customer subscribes to streaming movies.	Categorical
1	Contract	Type of service contract the customer holds.	Categorical

6			
17	Paperless_Billing	Whether the customer uses paperless billing.	Categorical
18	Payment_Method	Payment method used by the customer.	Categorical
19	Monthly_Charges	Monthly amount billed to the customer	Numeric (continuous)
20	Total_Charges	Total cumulative charges billed to the customer.	Numeric (continuous)
21	Churn	Whether the customer has churned.	Categorical (binary)

The primary objective of this project is to predict customer churn, defined as the target variable indicating whether a customer has discontinued service (yes) or continued to do business with the company (no). As shown in Figure 1, the dataset contains 5,174 non-churn customers and 1,869 churn customers.

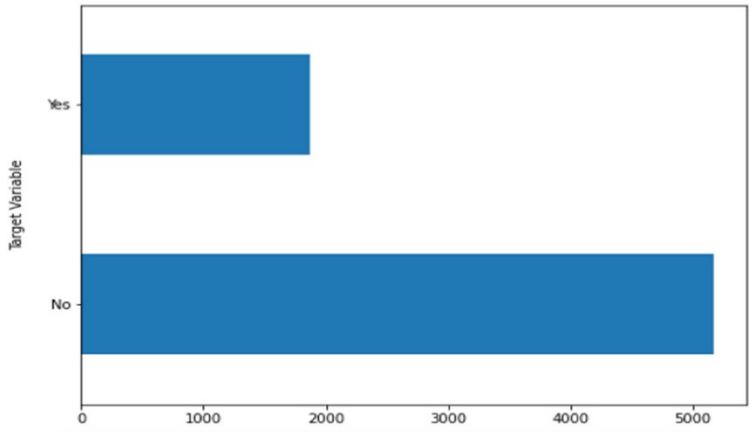


Figure 1: Distribution of the Target Variable (Churn)

Steps to apply Algorithms

The overall process followed in this project is summarized in Figure 1. It presents the sequential steps from data set import, data cleaning and exploratory analysis to model building and evaluation using logistic regression, support vector machine (SVM) and XGBoost.

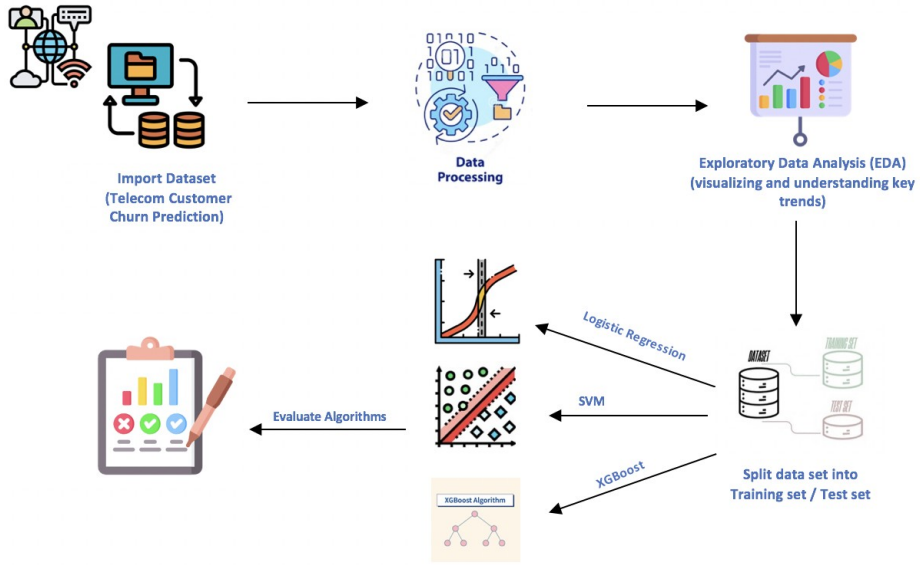


Figure 2: Project workflow

1. Data Preprocessing

After importing the dataset from Cagle into R Cloud, the dataset was cleaned by handling missing values. Specifically, we identified 11 rows with missing values in the Total_Charges column. Upon examination, it was also found that these rows had a duration = 0, indicating that these were newly joined customers who had not yet been billed. Since the number of affected rows was small and did not significantly affect the size of the dataset, they were removed to preserve data integrity.

Additionally, the Total_Charges column was originally stored as character data due to some blank entries. After handling missing values, the column was converted to a numeric format using the as.numeric () function combined with trimws() to remove leading/trailing spaces.

1.1 Check Structure and Missing Values

```
str(df)
summary(df)
colSums(is.na(df))       # Count NA values per column
```

```
> colSums(is.na(df))       # Count NA values per column
      customerID           gender    SeniorCitizen          Partner
               0                0                0                0
      Dependents           tenure     PhoneService    MultipleLines
               0                0                0                0
 InternetService   OnlineSecurity     OnlineBackup DeviceProtection
               0                0                0                0
     TechSupport      StreamingTV  StreamingMovies         Contract
               0                0                0                0
PaperlessBilling    PaymentMethod   MonthlyCharges     TotalCharges
               0                0                0               11
           Churn
```

Figure 3: Checking Data Structure and Identifying Missing Values in the Dataset

1.2 Show Missing Values as a Data Frame

	Column	MissingCount	MissingPercent
customerID	customerID	0	0.00
gender	gender	0	0.00
SeniorCitizen	SeniorCitizen	0	0.00
Partner	Partner	0	0.00
Dependents	Dependents	0	0.00
tenure	tenure	0	0.00
PhoneService	PhoneService	0	0.00
MultipleLines	MultipleLines	0	0.00
InternetService	InternetService	0	0.00
OnlineSecurity	OnlineSecurity	0	0.00
OnlineBackup	OnlineBackup	0	0.00
DeviceProtection	DeviceProtection	0	0.00
TechSupport	TechSupport	0	0.00
StreamingTV	StreamingTV	0	0.00
StreamingMovies	StreamingMovies	0	0.00
Contract	Contract	0	0.00
PaperlessBilling	PaperlessBilling	0	0.00
PaymentMethod	PaymentMethod	0	0.00
MonthlyCharges	MonthlyCharges	0	0.00
TotalCharges	TotalCharges	11	0.16
Churn	Churn	0	0.00

Figure 4: Missing Value in Dataset

Figure 3 shows that the dataset is nearly complete, with missing values found only in the Total_Charges column (11 missing entries, approximately 0.16%).

1.3 Visualize Missing Data

Figure 2 shows the distribution of missing values across all variables in the dataset using a missing values map. The red lines indicate missing entries, while the grey area represents observed (non-missing) data. As shown, missing data only occurs in the Total_Charges column, affecting a small portion of the dataset (11 rows out of 7043). All other columns are complete, indicating that the data set is mostly clean and requires only minimal processing of missing values. This visualization confirms the previous summary and supports the decision to remove these few rows during preprocessing to maintain data integrity.

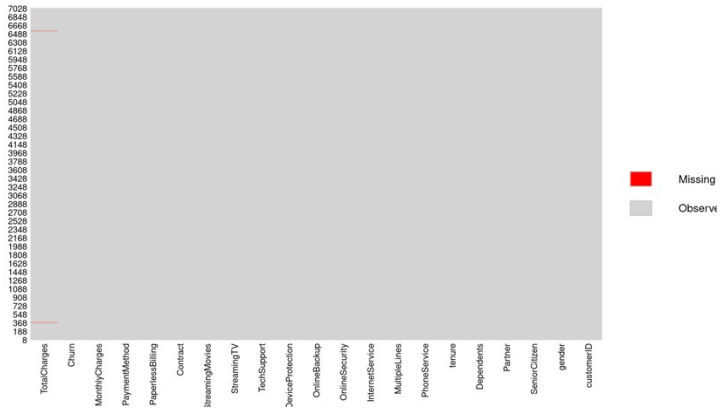


Figure 5: Visualization of missing data in the dataset

Rows with missing Total_Charges values (only 11) were removed since they corresponded to new customers with tenure = 0, indicating that these were newly joined customers who had not yet been billed.

1.4 Data cleaning

The Customer_ID variable, which serves solely as a unique identifier for each customer, was dropped from the dataset prior to model training, as it carries no predictive value and does not contribute to the classification task.

2. Exploratory Data Analysis (EDA)

Exploratory data analysis (EDA) was performed to reveal patterns and relationships between customer attributes and withdrawal behavior. Both categorical and numerical variables were conceptualized to identify important trends, such as high churn rates between short-term clients and those with monthly contracts, that guided feature selection and model interpretation.

2.1 Visualizing the Correlation

To understand the relationship between customer attributes and the likelihood of churn, we visualized both categorical and numerical features against the target variable "Churn". This step allowed us to identify patterns and feature behaviors that distinguish churned customers from those who stayed.

2.1.1 Visualizing categorical Features:

Stacked bar charts used to compare the proportion of churn (Yes) and non-churn (No) customers across different categories such as Contract_Type, Internet_Service, and Payment_Method as shown in *Figure 3*, *Figure 4* and *Figure 5*. These visualizations helped in identifying categorical variables with noticeable churn differences.

- **Churn by Contract Type**

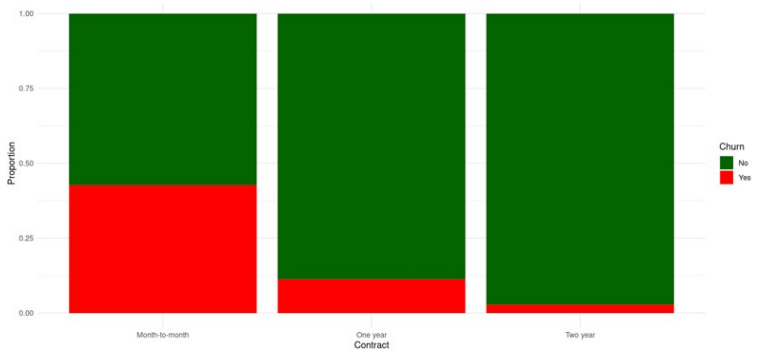


Figure 6: Churn rate by type of contract

The above graph (Figure 5) shows the proportion of customers who have churned and those who have not churned across three different contract types: month-to-month, one-year and two-year. Red syllables represent customers who have stopped using the service ("yes"), while green syllables represent customers who have stayed ("no").

From the graph, it is clear that customers on monthly contracts have a much higher rate of customer loss compared to those on one - or two-year contracts, whose customer loss rates are much lower. This suggests that customers with longer contractual obligations are more likely to remain loyal, highlighting contract type as an important predictor of withdrawal behavior.

- **Churn by Internet Services**

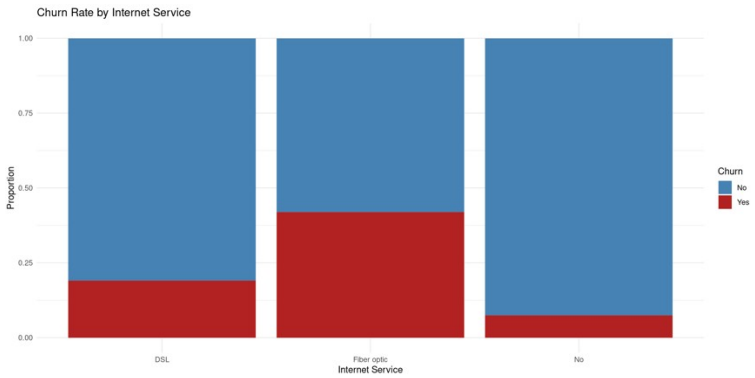


Figure 7:Churn rate by Internet services

This stacked bar graph shows the distribution of customer churn across different types of Internet services: DSL, fiber optic, and no Internet service. The red part represents customers who have churn ("yes") and the blue part indicates those who have not churn ("no").

Visualization reveals that customers with fiber optic internet have the highest customer loss rate, while those without internet service have the lowest. Customers using DSL fall in the middle. This suggests that the type of Internet service affects customer satisfaction and retention, possibly due to different performance, prices or quality of service.

Therefore, users of optical fibers may need more involvement or tailored retention strategies.

2.1.2 Visualizing Numerical Features

To better understand the distribution of numerical features across customers who churned (Churn = Yes) and those who did not (Churn = No), we used density plots**.** These plots allowed us to examine how values of each numeric feature differ between the two churn classes.

- **Tenure**

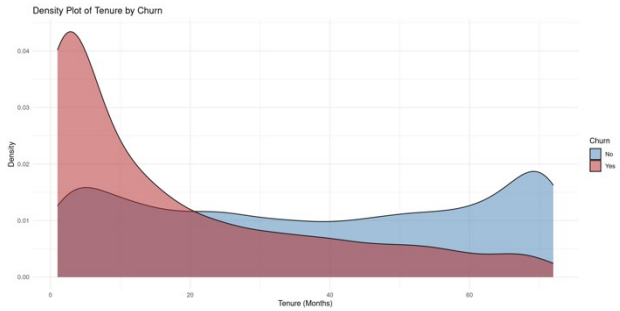


Figure 8: Density Plot of Tenure by Churn

This density plot compares the distribution of customer service duration (in months) between churned customers (red) and non-churned customers (blue). The graph shows a clear pattern:

- Customers who discontinue a subscription tend to have a shorter subscription duration, with a peak in the first months of their subscription.
- By contrast, customers with a longer service life (over 30 months) are less likely to withdraw, as indicated by the higher density in the blue curve.

This suggests a strong negative correlation between length of service and rate of customer loss, meaning that new customers are more likely to leave, while long-term customers tend to remain loyal. These insights underscore the importance of early engagement and employee retention strategies.

- **Monthly Charges**

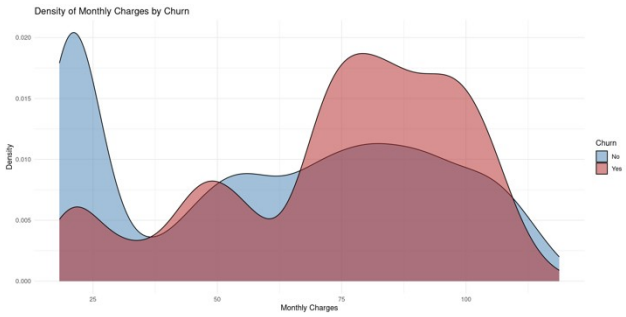


Figure 9: Density Plot of Monthly Charges by Churn

This density plot illustrates how monthly charges differ between customers who churned (red) and those who did not (blue). The red curve shows that churned customers are more concentrated at higher monthly charges, particularly between $70 and $100.In contrast, non-churned customers tend to have lower monthly charges, especially under $30. This indicates that customers paying more per month are at a greater risk of churning, possibly due to pricing dissatisfaction or perceived lack of value. These insights suggest that pricing strategies and value-added services should be carefully managed to retain high-paying customers and reduce churn.

- **Total Charges**

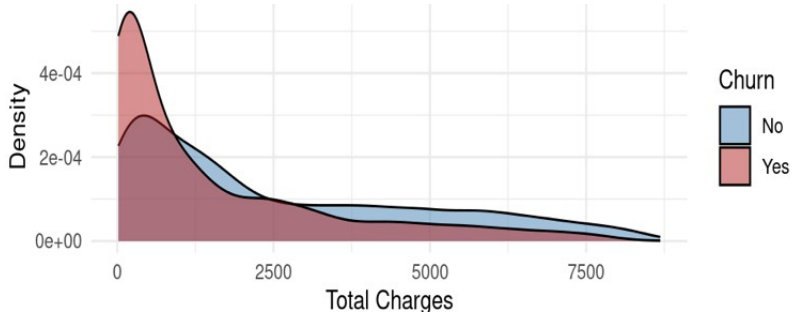


Figure 10: Density Plot of Total Charges by Churn

This density plot compares the distribution of total charges between churned customers (red) and non-churned customers (blue). The chart

shows that customers who churned generally have lower total charges, while customers with higher total charges are more likely to have remained with the company.

This aligns with the earlier observation from the tenure plot, as total charges are often a reflection of how long a customer has been subscribed. The findings suggest that short-term, low-paying customers are at higher risk of churning, reinforcing the importance of improving early customer experience and promoting long-term value.

3. Applying Three Machine Learning Algorithms for Churn Prediction

This section evaluates the ability of three classification algorithms to predict customer churn. We implemented:

1. Logistic Regression Algorithm
2. Support Vector Machine (SVM) Algorithm
3. XGBoost Algorithm

Each model was trained on the same dataset partitioning and evaluated using rating scales such as accuracy adjustment, confusion and recall matrices. The aim is to compare their predictive performance and identify the most appropriate algorithm to support business retention strategies.

3.1 Data Preparation

To start the modelling process, the necessary R packages were loaded to support data processing, pre-processing, and algorithm execution. The tidyverse package was used for data processing and transformation, Caret provided tools for model training and evaluation, xgboost enabled implementation of the XGBoost algorithm, and e1071 supported use of the Support Vector Machine (SVM).

The target variable "*Churn*" was then considered to ensure compatibility with the classification models, allowing the algorithms to correctly interpret the Churn condition as a categorical result.

Then, the data set was divided into training and test groups using a 70/30 split. Stratified sampling was applied to maintain the original ratio of churn and non-churn clients in both groups, ensuring a reliable and fair assessment during the model testing.

For clarity and to preserve the readability of the main report, all R code associated with data preparation, model implementation, and assessment procedures is provided in the Appendix.

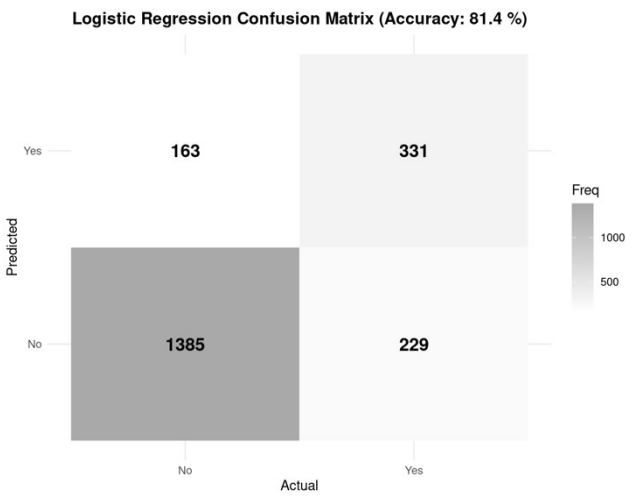
3.2 Machine Learning Model Evaluations and Predictions:

1. Logistic Regression

Table 2: Logistic Regression: Confusion Matrix Output and Heatmap Visualization

Output	Confusion matrix heatmap

```
Confusion Matrix and Statistics

          Reference
Prediction   No  Yes
       No  1385  229
       Yes  163  331

               Accuracy : 0.814
                 95% CI : (0.7968, 0.8304)
    No Information Rate : 0.7343
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5048

 Mcnemar's Test P-Value : 0.001027

            Sensitivity : 0.8947
            Specificity : 0.5911
         Pos Pred Value : 0.8581
         Neg Pred Value : 0.6700
             Prevalence : 0.7343
         Detection Rate : 0.6570
   Detection Prevalence : 0.7657
      Balanced Accuracy : 0.7429

       'Positive' Class : No
```

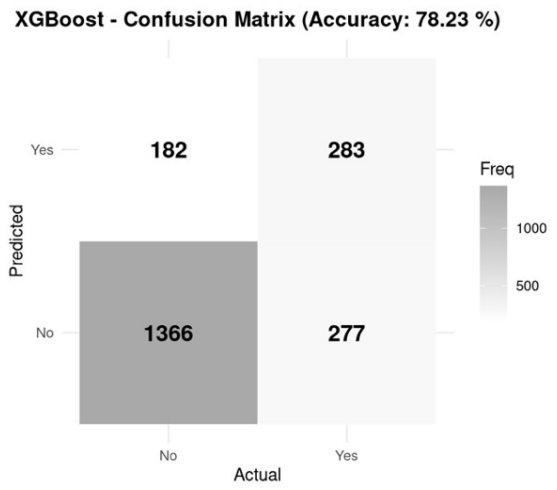


The confusion matrix for the Logistic Regression model shows an overall accuracy of 81.4%, indicating that the model correctly predicted the churn status of customers in the majority of cases. The sensitivity (recall for the non-churn class) is high at 89.47%, meaning the model is effective at identifying customers who are likely to stay. However, the specificity is lower at 59.11%, indicating a moderate ability to correctly identify churned customers.

Overall, Logistic Regression provides a strong baseline model, with good generalization ability, particularly in detecting non-churned customers, although its ability to detect churners could be improved.

2. XGBoost

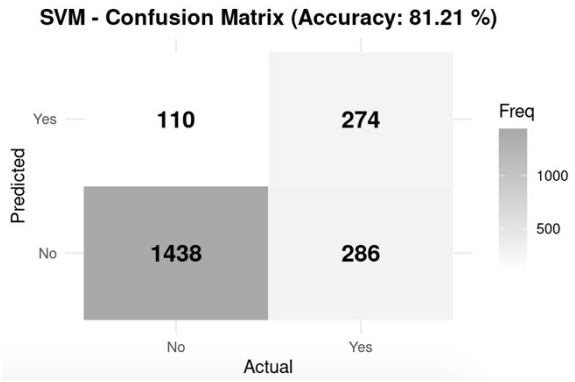
Table 3: XGBoost: Confusion Matrix Output and Heatmap Visualization

Output	Confusion matrix heatmap
XGBoost Confusion Matrix: > print(xgb_conf) Confusion Matrix and Statistics Reference Prediction No Yes No 1366 277 Yes 182 283 Accuracy : 0.7823 95% CI : (0.764, 0.7997) No Information Rate : 0.7343 P-Value [Acc > NIR] : 2.133e-07 Kappa : 0.41 Mcnemar's Test P-Value : 1.146e-05 Sensitivity : 0.8824 Specificity : 0.5054 Pos Pred Value : 0.8314 Neg Pred Value : 0.6086 Prevalence : 0.7343 Detection Rate : 0.6480 Detection Prevalence : 0.7794 Balanced Accuracy : 0.6939 'Positive' Class : No	

The XGBoost model achieved an overall accuracy of 78.23%, slightly lower than logistic regression, but still showed strong classification performance. The sensitivity (true positive rate for non-churn) is 88.24%, demonstrating the effectiveness of the model in identifying customers who stay. Overall, XGBoost provides a conservative balance between recall and accuracy, and while it performs slightly less well than logistic regression in terms of overall accuracy and kappa, the model remains applicable especially in contexts where minimizing false positives is important.

3. Support Vector Machine (SVM)

Table 4: Support Vector Machine (SVM): Confusion Matrix Output and Heatmap Visualization

Output	Confusion matrix heatmap
Confusion Matrix and Statistics Reference Prediction No Yes No 1438 286 Yes 110 274 Accuracy : 0.8121 95% CI : (0.7948, 0.8286) No Information Rate : 0.7343 P-Value [Acc > NIR] : < 2.2e-16 Kappa : 0.4648 Mcnemar's Test P-Value : < 2.2e-16 Sensitivity : 0.9289 Specificity : 0.4893 Pos Pred Value : 0.8341 Neg Pred Value : 0.7135 Prevalence : 0.7343 Detection Rate : 0.6822 Detection Prevalence : 0.8178 Balanced Accuracy : 0.7091 'Positive' Class : No	

The Support Vector Machine (SVM) model achieved an overall accuracy of 81.21%, closely matching the performance of Logistic Regression. It recorded the highest sensitivity among all three models at 92.89%, indicating strong effectiveness in identifying customers who will not churn.

Overall, SVM demonstrates strong recall, making it especially valuable in contexts where identifying churn risk is critical, even if it results in more false positives. This makes it a suitable choice for proactive customer retention strategies, where the cost of missing a churner is higher than the cost of incorrectly flagging a loyal customer.

Discussion and Results

This section presents and interprets the results of three machine learning models applied to the Telco Customer Churn Dataset، Logistic Regression, Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost):

- Logistic regression was chosen because it is easy to understand and works well for problems with two possible outcomes, such as volatility or non-volatility. It also helps explain how different features affect a customer's chance of leaving.
- The Support Vector Machine (SVM) was used because it could handle more complex patterns in the data. This system works by finding the best boundaries between customers who stop doing business with customers and those who don't, even if the data is not separated.
- XGBoost was chosen for its strong performance and ability to work well with structured data. It uses many decision trees to make accurate predictions and is useful when relationships between features are more complex.

Models were trained on 70% of the data and tested on the remaining 30%. The assessment scales included accuracy, precision, recall and specificity, all derived from confusion matrices. The results of the models shown in the below table.

Table 5: Performance Comparison of ML algorithms for Customer Churn Prediction

	Logistic Regression	SVM	XGBoost
Accuracy	81.40%	81.21%	78.23%
Sensitivity (Recall)	89.47%	92.89%	88.24%
Specificity	59.11%	48.93%	50.54%
Precision (PPV)	85.81%	83.41%	83.14%
Balanced Accuracy	74.29%	70.91%	69.39%

Among the three models, Logistic Regression demonstrated the most balanced performance, achieving the highest accuracy (81.40%), precision (85.81%), and recall (59.11%), as well as the highest balanced accuracy (74.29%). This indicates the effectiveness of logistic regression in identifying both lapsed and non-lapsed customers, with minimal false alarms. Its interpretability also supports its suitability for practical use, where clear and interpretable outputs are essential for management confidence and adoption.

Although SVM had a slightly lower accuracy (81.21%), it achieved the highest sensitivity (92.89%), demonstrating its strength in identifying lapsed customers. This makes it highly valuable in environments where the cost of lapsed customers outweighs the cost of mistargeting. However, SVM's lower accuracy (48.93%) indicates a high rate of false positives, which can lead to ineffective customer engagement strategies.

XGBoost offered a moderate balance, with average recall (88.24%) and precision (83.14%), but it had the lowest accuracy (78.23%) of the three. Its adaptability and robustness still make it a strong candidate for automated systems or high-dimensional datasets, where scalability and periodic model updates are priorities.

Interpretation of Results

These results illustrate the fundamental trade-offs between different classification algorithms. Logistic Regression provides a balance between recall and specificity, and its interpretability is highly desirable in business contexts where decision transparency is required. SVM, on the other hand, favors recall over specificity, making it an appropriate choice when the cost of missing a churner outweighs the consequences of false alarms. Meanwhile, XGBoost may be more appropriate in automated systems where continuous learning and streaming data are present, and where model flexibility and regularization capabilities are beneficial.

The variation in performance also underscores the importance of evaluating classifiers using a multidimensional set of metrics. Accuracy alone may be misleading in imbalanced classification tasks such as churn prediction, where the number of non-churners typically outweighs churners. Therefore, metrics such as balanced accuracy may provide more robust indicators of model efficacy.

Business Implications

The results of this project have practical relevance for organizations seeking to improve customer retention through predictive analytics. Among the three models evaluated, logistic regression stood out as the most effective in terms of overall accuracy and balanced performance. Their interpretability and relatively strong recall make them well-suited for operational deployment, particularly in business contexts where understanding the drivers of volatility is essential.

These findings align with existing literature emphasizing the importance of predictive analytics in customer retention strategies. For instance, Manzoor et al. (2023) highlight the critical role of machine learning in anticipating customer churn and guiding business decisions to enhance profitability.

From a business perspective, recall (sensitivity) is a critical metric, as failure to identify a customer at risk of losing customers may result in

lost revenue, missed cross-selling opportunities, and increased acquisition costs. While all three models showed reasonable accuracy, their recall scores suggest that additional measures may be needed to maximize true positive detection of churners.

Organizations can use the output of these models to segment customers based on the risk of losing customers and design targeted retention campaigns. For example, personal incentives, service upgrades, or loyalty rewards may be offered to high-risk customers to increase their likelihood of retention. Studies have shown that predictive customer loss models integrated into customer relationship management (CRM) systems can significantly reduce customer loss rates and improve marketing ROI (Suh, 2023).

Beyond prediction, the study reinforces the importance of investing in customer loyalty initiatives. The negative correlation between tenure and churn as shown in Figure 11 suggests that long-term customers are significantly less likely to leave, highlighting the value of loyalty cultivation. Companies can act on this by implementing programs such as:

- Personalized loyalty rewards
- Exclusive service benefits
- Long-term contract incentives
- Consistent engagement throughout the customer lifecycle

These approaches not only reduce churn risk but also increase customer lifetime value. In addition, companies should consider prioritizing touchpoints in early life cycle and inclusion processes, as new customers tend to be more likely to drop out in the first few months (Liu et al., 2024)

Limitations

Although the results of this project are promising, there are some limitations to consider:

- Class Imbalance: The dataset contains a larger number of non-churners than churners, which may affect the models' ability to detect churn. While balanced accuracy was used to mitigate this issue, future research could apply techniques such as oversampling or SMOTE to improve fairness.
- Static Dataset: Models were trained on historical data, meaning they may not perform as well if customer behavior changes in the future. This highlights the need for periodic model updates.
- Model Complexity: Although models like XGBoost are powerful, they are also more difficult to interpret. This may limit their use in business settings where managers prefer clear explanations for predictions.

Recommendations

Based on the model results and business context, the following recommendations are made:

- Use logistic regression as a baseline model for predicting churn due to its balanced performance and high interpretability, which supports business decision-making.
- Apply the SVM algorithm when the primary goal is to detect as many customers as possible who may be at risk of churn, especially in cases where preventing churn is more important than reducing false positives.
- Consider using XGBoost in future projects involving large datasets or requiring frequent retraining, especially when integrated into automated systems.
- Balance the dataset using resampling techniques to help the model learn better from both churn and non-churn customers.
- Regularly retrain models with updated customer data to reflect changes in customer behavior and ensure model effectiveness.

Conclusion

This project assessed the effectiveness of three machine learning algorithms: Logistic Regression, Support Vector Machine, and XGBoost, in predicting customer churn. Logistic Regression demonstrated the most balanced and interpretable performance, achieving high accuracy and precision. SVM was most effective in identifying churners due to its high recall, while XGBoost offered a conservative trade-off between recall and precision, albeit with slightly lower overall performance.

The study also revealed a strong negative relationship between customer service duration and customer churn rate, indicating that repeat customers are much less likely to leave. This finding highlights the importance of maintaining customer engagement and loyalty over time.

Overall, the findings support utilizing machine learning as a valuable tool for churn prediction. However, model selection should consider not only predictive accuracy, but also business priorities, the cost of misclassification, and the need for interpretability in practical applications.

References

Hadden, J., Tiwari, A., Roy, R., & Ruta, D. (2007). Computer assisted customer churn management: State-of-the-art and future trends. Computers & Operations Research, 34(10), 2902-2917.

Liu, X., Xia, G., Zhang, X., Ma, W., & Yu, C. (2024). Customer churn prediction model based on hybrid neural networks. Scientific Reports, 14(1). https://doi.org/10.1038/s41598-024-79603-9

Manzoor, A., Qureshi, M. A., Kidney, E., & Longo, L. (2024). A review on machine learning methods for customer churn prediction and recommendations for business practitioners. IEEE access.

Suh, Y. (2023). Machine learning based customer churn prediction in home appliance rental business. Journal of Big Data, 10(1). https://doi.org/10.1186/s40537-023-00721-8

Shaikh, A. K., Nazir, A., Khan, I., & Shah, A. S. (2022). Short term energy consumption forecasting using neural basis expansion analysis for interpretable time series. *Scientific reports*, 12(1), 22562.

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).

ybifoundation. (2024, February 28). Telecom customer churn prediction. Kaggle. https://www.kaggle.com/code/ybifoundation/telecom-customer-churn-prediction/notebook

Tessitore, S. (2024, April 18). *What's the average churn rate by industry?* CustomerGauge. https://customergauge.com/blog/average-churn-rate-by-industry

Appendix A.

R Code for Data Preparation, Modeling, and Evaluation

The full R code used for data cleaning, exploratory data analysis, model implementation, and performance evaluation is provided below to support transparency and reproducibility of the results presented in this report.

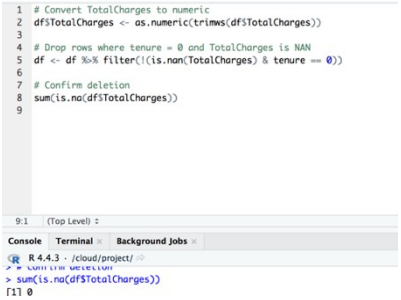
Loading libraries

```r
# Load necessary libraries
library(tidyverse)
library(data.table)
library(Amelia)
library(naniar)
library(VIM)
library(plotly)

# Suppress warnings
options(warn = -1)

```

Explore Data Structure

```r
str(df)
summary(df)
glimpse(df)

```

Visualizing the Correlation



```
# Convert TotalCharges to numeric
df$TotalCharges <- as.numeric(trimws(df$TotalCharges))

# Drop rows where tenure = 0 and TotalCharges is NAN
df <- df %>% filter(!(is.nan(TotalCharges) & tenure == 0))

# Confirm deletion
sum(is.na(df$TotalCharges))
```

```
> sum(is.na(df$TotalCharges))
[1] 0
```

Visualize Missing Data

```
library(Amelia)

missmap(df,
        main = "Missing Data Map - Amelia",
        col = c("red", "lightgrey"),
        legend = TRUE)
```

Data cleaning

```
library(dplyr)

df <- df %>% select(-customerID)

```

Churn by Contract Type

```
library(ggplot2)
library(tidyverse)

# Churn by Contract Type
ggplot(df, aes(x = Contract, fill = Churn)) +
  geom_bar(position = "fill") +
  labs(title = "Churn Rate by Contract Type", y = "Proportion", x = "Contract") +
  scale_fill_manual(values = c("No" = "darkgreen", "Yes" = "red")) +
  theme_minimal()
```

Churn by Contract Type

```
library(ggplot2)
library(tidyverse)


# Churn by Internet Service
ggplot(df, aes(x = InternetService, fill = Churn)) +
  geom_bar(position = "fill") +
  labs(title = "Churn Rate by Internet Service", y = "Proportion", x = "Internet Service") +
  scale_fill_manual(values = c("No" = "steelblue", "Yes" = "firebrick")) +
  theme_minimal()
```

Churn by Internet Services

```
library(ggplot2)
library(tidyverse)


# Churn by Internet Service
ggplot(df, aes(x = InternetService, fill = Churn)) +
  geom_bar(position = "fill") +
  labs(title = "Churn Rate by Internet Service", y = "Proportion", x = "Internet Service") +
  scale_fill_manual(values = c("No" = "steelblue", "Yes" = "firebrick")) +
  theme_minimal()
```

Monthly charge by Churn

```r
library(ggplot2)
library(tidyverse)
# Density plot for MonthlyCharges
ggplot(df, aes(x = MonthlyCharges, fill = Churn)) +
  geom_density(alpha = 0.5) +
  labs(title = "Density of Monthly Charges by Churn", x = "Monthly Charges", y = "Density") +
  scale_fill_manual(values = c("No" = "steelblue", "Yes" = "firebrick")) +
  theme_minimal()
```

Total charge by Churn

```r
library(ggplot2)
library(tidyverse)

ggplot(df, aes(x = TotalCharges, fill = Churn)) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Plot of Total Charges by Churn",
       x = "Total Charges", y = "Density") +
  scale_fill_manual(values = c("No" = "steelblue", "Yes" = "firebrick")) +
  theme_minimal()
```

Tenure by Churn

```r
library(ggplot2)
library(tidyverse)
ggplot(df, aes(x = tenure, fill = Churn)) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Plot of Tenure by Churn",
       x = "Tenure (Months)", y = "Density") +
  scale_fill_manual(values = c("No" = "steelblue", "Yes" = "firebrick")) +
  theme_minimal()
```

Applying Algorithm:

4.1.1 load the essential R packages:

```r
# Load Required Libraries

library(tidyverse)
library(caret)
library(xgboost)
library(e1071)
```

- tidyverse for data manipulation,
- caret for preprocessing and modeling support
- xgboost for gradient boosting implementation
- e1071 for Support Vector Machine (SVM)

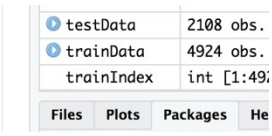
4.1.2 Convert Churn to a factor for classification:

```r
df$Churn <- as.factor(df$Churn)
```

4.1.3 Train-Test Split

Splits the dataset into 70% training and 30% testing subsets using stratified sampling to maintain the original distribution of churned vs. non-churned customer

```
library(caret)
set.seed(123)
df$Churn <- as.factor(df$Churn)
trainIndex <- createDataPartition(df$Churn, p = 0.7, list = FALSE)
trainData <- df[trainIndex, ]
testData <- df[-trainIndex, ]
```



4.1.4 Define Heatmap Plot Function

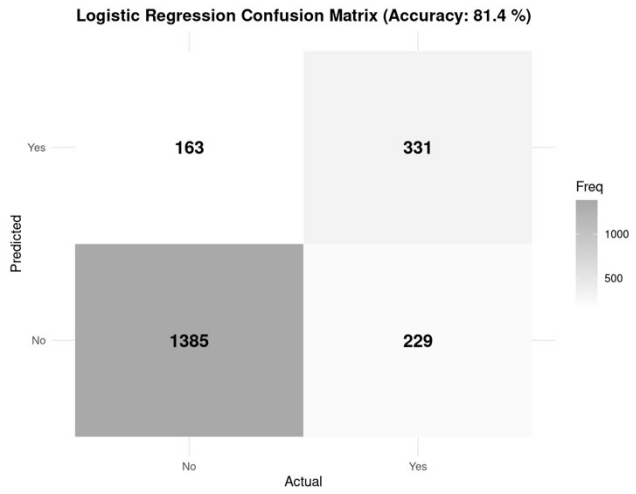
```
#Define Heatmap Plot Function
plot_conf_matrix <- function(conf_matrix, model_name) {
  cm_df <- as.data.frame(as.table(conf_matrix$table))
  colnames(cm_df) <- c("Prediction", "Reference", "Freq")
  acc <- round(conf_matrix$overall['Accuracy'] * 100, 2)

  p <- ggplot(cm_df, aes(x = Reference, y = Prediction, fill = Freq)) +
    geom_tile(color = "white") +
    geom_text(aes(label = Freq), size = 5, fontface = "bold") +
    scale_fill_gradient(low = "white", high = "darkgrey") +
    labs(title = paste(model_name, "- Confusion Matrix (Accuracy:", acc, "%)"),
         x = "Actual", y = "Predicted") +
    theme_minimal() +
    theme(plot.title = element_text(hjust = 0.5, face = "bold"))

  print(p)
}
```

Logistic Regression

R Code	Output	Confusion matrix heatmap

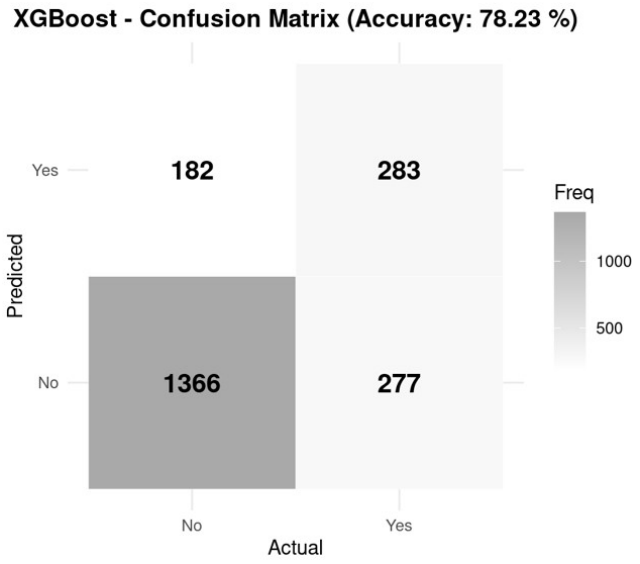
```r
# Logistic Regression

log_model <- glm(Churn ~ ., data = trainData, family = binomial)
log_prob <- predict(log_model, newdata = testData, type = "response")
log_pred <- factor(ifelse(log_prob > 0.5, "Yes", "No"), levels = c("No", "Yes"))
log_true <- factor(testData$Churn, levels = c("No", "Yes"))
log_conf <- confusionMatrix(log_pred, log_true)

cat("Logistic Regression\n")
print(log_conf)
plot_conf_matrix(log_conf, "Logistic Regression")
```

```
Confusion Matrix and Statistics

          Reference
Prediction   No  Yes
       No  1385  229
       Yes  163  331

               Accuracy : 0.814
                 95% CI : (0.7968, 0.8304)
    No Information Rate : 0.7343
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5048

 Mcnemar's Test P-Value : 0.001027

            Sensitivity : 0.8947
            Specificity : 0.5911
         Pos Pred Value : 0.8581
         Neg Pred Value : 0.6700
             Prevalence : 0.7343
         Detection Rate : 0.6570
   Detection Prevalence : 0.7657
      Balanced Accuracy : 0.7429

       'Positive' Class : No
```



XGBoost

R Code	Output	Confusion matrix heatmap

`xgb_pred <- ifelse(xgb_prob > 0.5, "Yes", "No")`
`xgb_pred <- factor(xgb_pred, levels = c("No", "Yes"))`
`xgb_true <- factor(testData$Churn, levels = c("No", "Yes"))`
`xgb_conf <- confusionMatrix(xgb_pred, xgb_true)`
 	
`cat("XGBoost Confusion Matrix:\n")`
`print(xgb_conf)`
	XGBoost Confusion Matrix:
	`> print(xgb_conf)`
	Confusion Matrix and Statistics
	
	Reference
	Prediction No Yes
	No 1366 277
	Yes 182 283
	
	Accuracy : 0.7823
	95% CI : (0.764, 0.7997)
	No Information Rate : 0.7343
	P-Value [Acc > NIR] : 2.133e-07
	
	Kappa : 0.41
	
	Mcnemar's Test P-Value : 1.146e-05
	
	Sensitivity : 0.8824
	Specificity : 0.5054
	Pos Pred Value : 0.8314
	Neg Pred Value : 0.6086
	Prevalence : 0.7343
	Detection Rate : 0.6480
	Detection Prevalence : 0.7794
	Balanced Accuracy : 0.6939
	
	'Positive' Class : No
		

R Code	Output	Confusion matrix heatmap